# **Interfacing Matrix Keypad with PIC Microcontroller**

**ABSTRACT**

Group members

1.Jagadeesh

2. Sai Kaushik

3.Mrudula

4.Chaithanya

5.Anandh

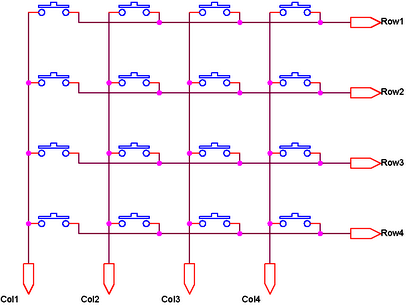
**PIC MICROCONTROLLER**

**PIC** is a family of modified Harvard architecture microcontrollers made by Microchip Technology, derived from the PIC1650 originally developed by General Instrument's Microelectronics Division. The name PIC initially referred to *Peripheral Interface Controller*. The first parts of the family were available in 1976; by 2013 the company had shipped more than twelve billion individual parts, used in a wide variety of embedded systems.

Early models of PIC had read-only memory (ROM) or field-programmable EPROM for program storage, some with provision for erasing memory. All current models use Flash memory for program storage, and newer models allow the PIC to reprogram itself. Program memory and data memory are separated. Data memory is 8-bit, 16-bit and in latest models, 32-bit wide. Program instructions vary in bit-count by family of PIC, and may be 12, 14, 16, or 24 bits long. The instruction set also varies by model, with more powerful chips adding instructions for digital signal processing functions.

**MATRIX KEYPAD**

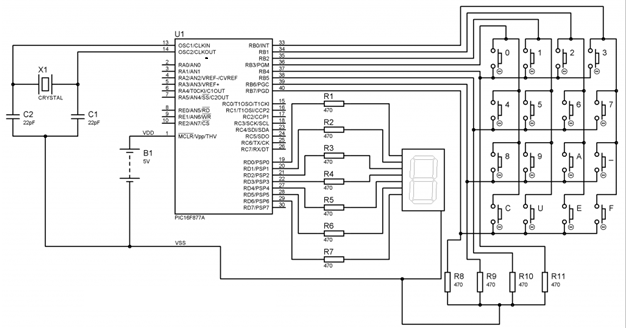
Matrix Keypad is a very useful and user friendly when we want to design certain applications like Calculator, Telephone etc. Matrix Keypad is made by arranging push button switches in rows and columns. Just imagine, if you want to interface a 4\*4 (16 keys) matrix keypad with a microcontroller.  In the straight forward way, you will need 16 pins of a microcontroller for that, but by using a simple technique we can reduce it to 8 pins. In the matrix keypad switches are connected in a special manner a shown in the figure below.



**CIRCUIT DIAGRAM**

Pressed keys can be detected by Scanning. All column connections (Col1 – Col4) are input pins and all row connections (Row1 – Row4) are output pins. In the normal case (not scanning) all column inputs where in LOW (GND) state. For scanning keypad,

1. A Logic HIGH signal is given to Col1 of column inputs.
2. Then each Row output (row1 – row4) is scanned one by one. If any of the key belongs to first column is pressed, the Logic high signal from the Col1 will pass to that row. Through we can detect the key.
3. This process is repeated for all columns if we want to detect multiple keys.

****

**Note:**VDD and VSS of the pic microcontroller is not shown in the circuit diagram. VDD should be connected to +5V and VSS to GND.

Matrix Keypad is connected to the PORTB of the PIC Microcontroller. Each column of the Matrix Keypad is connected to RB0 – RB3 of the PIC Microcontroller, which are configured as output pins. While each row of the Matrix Keypad is connected to RB4 – RB7 of the PIC Microcontroller, which are configured as input pins.

**WORKING**

Using 4\*4 matrix keypad, having characters 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, -, C, U, E, F. ‘B’ is replaced by ‘-‘ and ‘D’ is replaced by ‘U’ because Seven Segment Display is used for displaying characters. ‘B’ will be similar to ‘8’ and ‘D’ will be similar to ‘0’ when displayed in Seven Segment Display. For reading data for the Matrix Keypad, each column is made high and rows are scanned.

In the scanning algorithm, we test*Row 1* keys first by making *Row1*wire equal to zero and making other rows one. Then each column is checked if it is equal to zero. Then *Row2*wire is made zero, while making all other rows one. Then again each column is checked to detect if any *Row2*key is pressed. This pattern is repeated for every row. If no key is pressed, then 'n' value is returned from the function.  
  
For example, if key 7 was pressed when this code was being executed then *C1*will be zero ( Because keypad shorts the corresponding row and column wire ). If C1 equals to 0 condition will become true and there is a small delay of some micro seconds after which when C1 equals to 0 statement is present. This statement basically makes the controller to wait until the key is no longer pressed, to remove the possibility of reading one pressed key many times. After that, value of the key ( i-e '7' in this case ) is returned from the function.